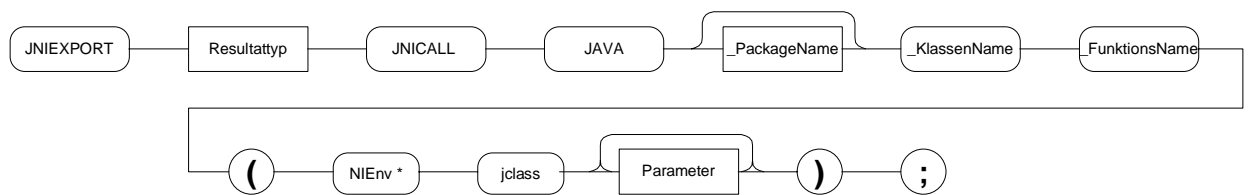


# Java Native Interface Parameterübergaben (JNI)

- Mit dem Java JNI kann hardwarenahe Programmierung praktiziert werden oder auch bereits bestehende C-Module weiterverwendet werden.
- Die extern aufzurufenden nativen Funktionen müssen in einer DLL liegen. Diese DLL muss die in JAVA zu verwendenden Funktionen in einer genau definierten Notation exportieren. Diese Informationen werden mit dem JDK-Tool javah.exe erzeugt. Direkt unterstützt ist aber meines Wissens nur die Sprache C/C++.
- Die über das JNI aufgerufenen Funktionen können parameterbehaftet sein und auch Resultate als Funktionswert zurückgeben.
- Der Funktionsprototyp der nativen Funktion wird durch javah.exe nach folgendem Format gebildet:



## Beispiel:

```
JNIEXPORT jint JNICALL Java_JavaBeispiele2_JNI1_addNumbers (JNIEnv *, jclass, jint, jint);
```

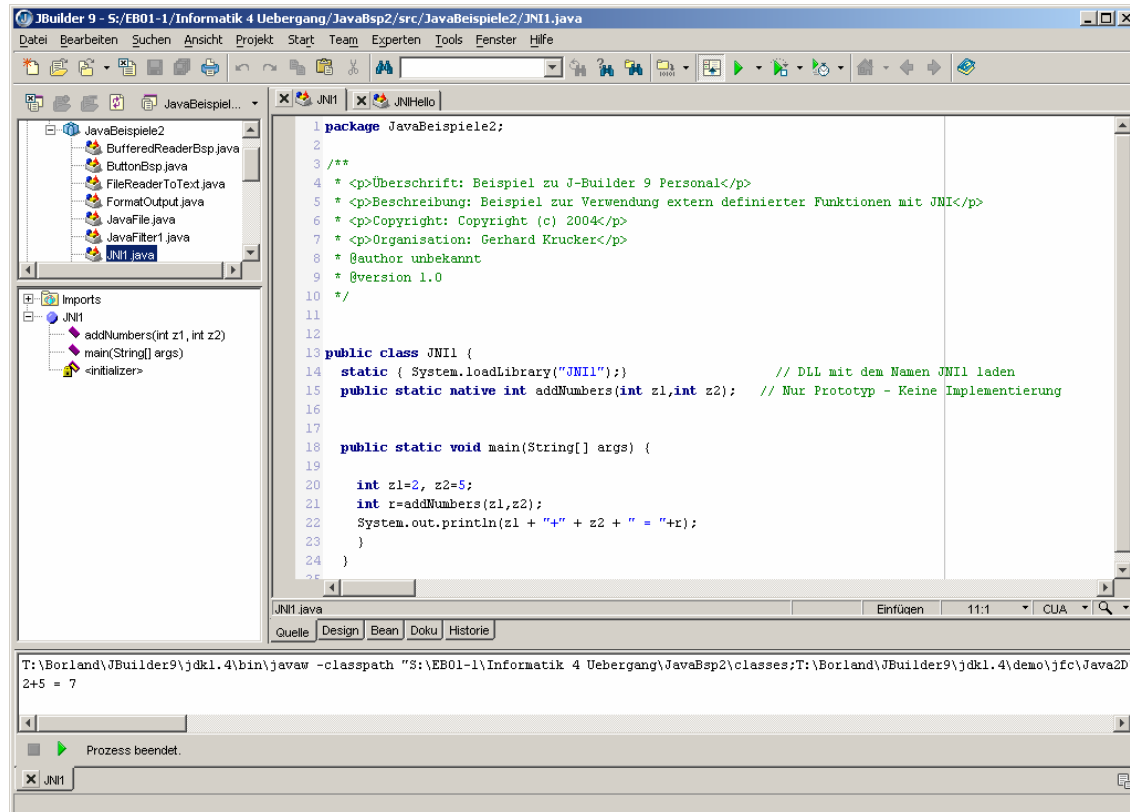
- Um Typenkompatibilität zu gewährleisten, definiert Java die Interfacedatentypen für Parameter:

Java Typ	Nativer Typ	Format
boolean	jboolean	8, unsigned
byte	jbyte	8
char	jchar	16, unsigned
short	jshort	16
int	jint	32
long	jlong	64
float	jfloat	32
double	jdouble	64
void	void	-

- Weiter sind über die Parameter `JNIEnv*` und `class` Zugriffe auf das Objekt, Attribute, oder die Klasse möglich.

Arbeitsbeispiel zu JNI mit Parameter und Funktionswertrückgabe:

## Java-Code:



## C-Builder-Projekt:

